A Simple Deterministic Reduction for the Gap Minimum Distance of Code Problem

Per Austrin*
University of Toronto

Subhash Khot†
New York University

October 8, 2010

Abstract

We present a simple deterministic gap-preserving reduction from SAT to the Minimum Distance of Code Problem over $\mathbb{F}_2$. We also show how to extend the reduction to work over any finite field. Previously a randomized reduction was known due to Dumer, Micciancio, and Sudan [8], which was recently derandomized by Cheng and Wan [6, 7]. These reductions rely on highly non-trivial coding theoretic constructions whereas our reduction is *elementary*.

As an additional feature, our reduction gives a constant factor hardness even for asymptotically good codes, i.e., having constant rate and relative distance. Previously it was not known how to achieve deterministic reductions for such codes.

*Research done while at New York University supported by NSF Expeditions grant CCF-0832795.

†Research supported by NSF CAREER grant CCF-0833228, NSF Expeditions grant CCF-0832795, and BSF grant 2008059.

1 Introduction

The Minimum Distance of Code Problem over a finite field $\mathbb{F}_q$, denoted MIN DIST(q), asks for a non-zero codeword with minimum Hamming weight in a given linear code C (i.e., a linear subspace of $\mathbb{F}_q^n$). The problem was proved to be NP-hard by Vardy [15].

Dumer, Micciancio, and Sudan [8] proved that assuming RP $\neq$ NP the problem is hard to approximate within some factor $\gamma > 1$ using a *gap preserving* reduction from the Nearest Codeword Problem, denoted NCP(q) (which is known to be NP-hard even with a large gap). The latter problem asks, given a code $\tilde{C} \subseteq \mathbb{F}_q^m$ and a point $p \in \mathbb{F}_q^m$, for a codeword that is nearest to p in Hamming distance. However, Dumer et al.'s reduction is randomized: it maps an instance $(\tilde{C}, p)$ of NCP(q) to an instance C of MIN DIST(q) in a randomized manner such that: in the YES Case, with high probability, the code C has a non-zero codeword with weight at most d, and in the NO Case, C has no non-zero codeword of weight less that γd, for some fixed constant $\gamma > 1$. We note that the minimum distance of code is multiplicative under the tensor product of codes; this enables one to *boost* the inapproximability result to any constant factor, or even to an *almost polynomial factor* (under a quasipolynomial time reduction), see [8].

The randomness in Dumer et al.'s reduction is used for constructing, as a gadget, a non-trivial coding theoretic construction with certain properties (see Section 1.1 for details). In a remarkable pair of papers, Cheng and Wan [6, 7] recently constructed such a gadget deterministically, thereby giving a deterministic reduction to the Gap MIN DIST(q) Problem. Cheng and Wan's construction is quite sophisticated. It is an interesting pursuit, in our opinion, to seek an *elementary* deterministic reduction for the Gap MIN DIST(q) Problem.

In this paper, we indeed present such a reduction. For codes over $\mathbb{F}_2$, our reduction is (surprisingly) simple, and does not rely on any specialized gadget construction. The reduction can be extended to codes over any finite field $\mathbb{F}_q$; however, then the details of the reduction becomes more involved, and we need to use Viola's recent construction of a psedorandom generator for low degree polynomials [16]. Even in this case, the resulting reduction is conceptuelly quite simple.

We also observe that our reduction produces asymptotically good codes, i.e., having constant rate and relative distance. While Dumer et al. [8] are able to prove randomized hardness for such codes, this was not obtained by the deterministic reduction by Cheng and Wan. In [7], proving a constant factor hardness of approximation for asymptotically good codes is mentioned as an open problem.

Our main theorem is thus:

Theorem 1.1. *For any finite field $\mathbb{F}_q$, there exists a constant $\gamma > 0$ such that it is NP-hard (via a deterministic reduction) to approximate the* MIN DIST(q) *problem to within a factor $1 + \gamma$, even on codes with rate $\geq \gamma$ and relative distance $\geq \gamma$ (i.e., asymptotically good codes).*

As noted before, the hardness factor can be boosted via tensor product of codes (though after a superconstant amount of tensoring the code is no longer asymptotically good):

Theorem 1.2. *For any finite field $\mathbb{F}_q$, and constant $\epsilon > 0$, it is hard to approximate the* MIN DIST(q) *problem to within a factor $2^{(\log n)^{1-\epsilon}}$ unless NP $\subseteq$ DTIME($2^{(\log n)^{O(1)}}$).*

Another motivation to seek a new deterministic reduction for MIN DIST(q) is that it might lead to a deterministic reduction for the analogous problem for integer lattices, namely the Shortest Vector Problem (SVP). For SVP, we do not know of a deterministic reduction that proves even the basic NP-hardness, let alone a hardness of approximation result. All known reductions are randomized [1, 5, 14, 11, 12, 10]. In fact, the reduction of Dumer et al. [8] giving hardness of approximation for MIN DIST(q) assuming NP $\neq$ RP is inspired by a reduction by Micciancio [14] for SVP.

Our hope is that our new reduction for MIN DIST(q) can be used to shed new light on the hardness of SVP. For instance, it might be possible to combine our reductions for MIN DIST(q) for different primes q so as to give a reduction over integers, i.e., a reduction to SVP.

1.1 Previous Reductions

On a high level, the idea of the reduction of Dumer et al. [8] is the following. We start from the hardness of approximation for NCP. Given an instance $(\tilde{C}, p)$, let us look at the code $C = \mathrm{span}(\tilde{C} \cup \{p\})$. Then any codeword of C which uses the point p must have large distance. However, it can be that the code $\tilde{C}$ itself has very small distance so that the minimum distance of C is unrelated to the distance from p to $\tilde{C}$. Loosely speaking, the idea is then to combine C with an additional code C' such that any codeword which does not use p must have a large weight in C'.

Let us briefly describe the gadget of [8]. They use a coding theoretic construction with the following properties (slightly restated). Let $\frac{1}{2} < \rho < 1$ be a fixed constant and k be a growing integer parameter. The field size q is thought of as a fixed constant.

1. $C^* \subseteq \mathbb{F}_q^\ell$ is a linear code with distance d, where ℓ is polynomial in k (think of $\ell = k^{100}$).
2. There is a "center" $v \in \mathbb{F}_q^\ell$ such that the ball of radius r around v, denoted $B(v, r)$, contains q^k codewords and $r = \lfloor \rho d \rfloor$. In notation, $|B(v, r) \cap C^*| \geq q^k$.
3. There is a linear map $T : \mathbb{F}_q^\ell \mapsto \mathbb{F}_q^{k'}$ such that the image of $B(v, r) \cap C^*$ under T is the full space $\mathbb{F}_q^{k'}$. Here k' is polynomial in k (think of $k' = k^{0.1}$).

Dumer et al. achieve such a construction in a randomized manner. They let C^* be a suitable concatenation of Reed-Solomon codes with the Hadamard code so that even a *typical* ball of radius r contains many (i.e., q^k) codewords. Hence choosing the center v at random satisfies the second property. They show further that a random linear map T satisfies the third property. By giving a deterministic construction of such a gadget, Cheng and Wan [6, 7] recently derandomized the reduction of [8].

1.2 Organization

We present a proof of this theorem for the binary field in Section 3 and for a general finite field in Section 5. Even for the binary case, it is instructive to first see a reduction to NCP(2) in Section 3.1 which is then extended to the MIN DIST(2) problem in Section 3.2.

2 Preliminaries

2.1 Codes

Let q be a prime power.

Definition 2.1. A linear code C over a field $\mathbb{F}_q$ is a linear subspace of $\mathbb{F}_q^n$, where n is the block-length of the code and dimension of the subspace C is the dimension of the code. The distance of the code $d(C)$ is the minimum Hamming weight of any non-zero vector in C.

The two problems Min Dist(q) and NCP(q) are defined as follows.

Definition 2.2. Min Dist(q) is the problem of determining the distance $d(C)$ of a linear code $C \subseteq \mathbb{F}_q^n$. The code may be given by the basis vectors for the subspace C or by the linear forms defining the subspace.

Definition 2.3. NCP(q) is the problem of determining the minimum distance from a given point $p \in \mathbb{F}_q^n$ to any codeword in a given code $C \subseteq \mathbb{F}_q^n$. Equivalently, it is the problem of determining the minimum Hamming weight of any point z in a given affine subspace of $\mathbb{F}_q^n$ (which would be $C - p$).

Our reduction uses tensor products of codes, which are defined as follows.

Definition 2.4. Let $C_1, C_2 \subseteq \mathbb{F}_q^n$ be linear codes. Then the linear code $C_1 \otimes C_2 \subseteq \mathbb{F}_q^{n^2}$ is defined as the set of all $n \times n$ matrices over $\mathbb{F}_q$ such that each of its columns is a codeword in C_1 and each of its rows is a codeword in C_2.

A well-known fact is that the distance of a code is multiplicative under the tensor product of codes.

Fact 2.5. Let $C_1, C_2 \subseteq \mathbb{F}_q^n$ be linear codes. Then the linear code $C_1 \otimes C_2 \subseteq \mathbb{F}_q^{n^2}$ has distance $d(C_1 \otimes C_2) = d(C_1)d(C_2)$.

We shall need the following Lemma which shows that for many codewords of $C \otimes C$ one can obtain a stronger bound on the distance than the bound $d(C)^2$ given by Fact 2.5.

Lemma 2.6. *Let $C \subseteq \mathbb{F}_q^n$ be a linear code of distance $d = d(C)$, and let $Y \in C \otimes C$ be a non-zero codeword with the additional properties that*

1. *The diagonal of Y is zero.*
2. *Y is symmetric.*

Then Y has at least $d^2(1 + 1/q)$ non-zero entries.

Proof. Suppose $Y_{ij} = Y_{ji} \neq 0$. Since we have $Y_{ii} = 0$ it must hold that $i \neq j$ and that rows i and j are linearly independent codewords of C. By Fact 2.7 below it follows that the number of columns k such that at least one of Y_{ik} and Y_{jk} is non-zero is at least $d(1 + 1/q)$. Each of these columns then has at least d non-zero entries and hence Y has at least $d^2(1 + 1/q)$ non-zero entries. □

Fact 2.7. Let $C \subseteq \mathbb{F}_q^n$ be a linear code of distance $d = d(C)$. Then for any two linearly independent codewords $x, y \in \mathbb{F}_q^n$, the number of coordinates $i \in [n]$ for which either $x_i \neq 0$ or $y_i \neq 0$ is at least $d(1 + 1/q)$.

Proof. Let m be the number of coordinates such that $x_i \neq 0$ or $y_i \neq 0$ but not both, and let m' be the number of coordinates such that both $x_i \neq 0$ and $y_i \neq 0$. Clearly,

$$m + 2m' \geq 2d.$$

We can choose $\lambda \neq 0$ appropriately so that the vector $x - \lambda y$ has at most $m + m' - m'/(q-1)$ non-zero entries. This implies

$$m + m' - m'/(q-1) \geq d.$$

Multiplying the first inequality by $1/q$, the second by $(q-1)/q$, and adding up gives $m + m' \geq d(1 + 1/q)$ as desired. □

2.2 Hardness of Constraint Satisfaction

The starting point in our reduction is a constraint satisfaction problem that we refer to as the Max NAND problem, defined as follows.

Definition 2.8. An instance Ψ of the Max NAND problem consists of a set of quadratic equations over $\mathbb{F}_2$, each of the form $x_k = \text{NAND}(x_i, x_j) = 1 + x_i \cdot x_j$ for some variables x_i, x_j, x_k. The objective is to find an assignment to the variables such that as many equations as possible are satisfied. We denote by $\mathsf{Opt}(\Psi) \in [0, 1]$ the maximum fraction of satisfied equations over all possible assignments to the variables.

The following is an easy consequence of the PCP Theorem [9, 3, 2] and the fact that NAND gates form a basis for the space of boolean functions.

Theorem 2.9. *There is a universal constant $\delta > 0$ such that given a* Max NAND *instance Ψ it is NP-hard to determine whether $\mathsf{Opt}(\Psi) = 1$ or $\mathsf{Opt}(\Psi) \leq 1 - \delta$.*

3 The Binary Case

In this section we give a simple reduction from Max NAND showing that it is NP-hard to approximate Min Dist(2) to within some constant factor.

3.1 Reduction to Nearest Codeword

It is instructive to start with a reduction for the Nearest Codeword Problem, NCP(2), for which it is significantly easier to prove hardness. There are even simpler reductions known than the one we give here, but as we shall see in the next section this reduction can be modified to give hardness for the Min Dist(2) problem.

Given a Max NAND instance Ψ with n variables and m constraints, we shall construct an affine subspace $\mathcal{S}$ of $\mathbb{F}_2^{4m}$ such that:

(i) If Ψ is satisfiable then $\mathcal{S}$ has a vector of Hamming weight at most m.

(ii) If $\mathsf{Opt}(\Psi) \leq 1 - 2\delta$ then $\mathcal{S}$ has no vector of Hamming weight less than $(1 + 2\delta)m$.

This proves, according to Definition 2.3, that NCP(2) is NP-hard to approximate within a factor $1 + 2\delta$.

Every constraint $x_k = 1 + x_i x_j$ in Ψ gives rise to four new variables, as follows. We think of the four variables as a function $S_{ijk} : \mathbb{F}_2^2 \to \mathbb{F}_2$. The intent is that this function should be the indicator function of the values of x_i and x_j, in other words, that

$$S_{ijk}(a, b) = \begin{cases} 1 & \text{if } x_i = a \text{ and } x_j = b \\ 0 & \text{otherwise} \end{cases} .$$

With this interpretation in mind, each function S_{ijk} has to satisfy the following linear constraints over $\mathbb{F}_2$:

$$\begin{aligned} S_{ijk}(0,0) + S_{ijk}(0,1) + S_{ijk}(1,0) + S_{ijk}(1,1) &= 1 && (1) \\ S_{ijk}(1,0) + S_{ijk}(1,1) &= x_i && (2) \\ S_{ijk}(0,1) + S_{ijk}(1,1) &= x_j && (3) \\ S_{ijk}(0,0) + S_{ijk}(0,1) + S_{ijk}(1,0) &= x_k. && (4) \end{aligned}$$

Thus, we have a set of $n+4m$ variables $z_1, \ldots, z_{n+4m}$ (recall that n and m are the number of variables and constraints of Ψ, respectively) and $4m$ linear constraints of the form $\sum l_{ij} z_j = b_i$ where $l_i \in \mathbb{F}_2^{n+4m}$ and $b_i \in \mathbb{F}_2$.

Let $\mathcal{S} \subseteq \mathbb{F}_2^{4m}$ be the affine subspace of $\mathbb{F}_2^{4m}$ defined by the set of solutions to the system of equations, projected to the $4m$ coordinates corresponding to the S_{ijk} variables. Note that these coordinates uniquely determine the remaining n coordinates (assuming without loss of generality that every variable of Ψ appears in some constraint), according to Equations (2)-(4).

Now, if Ψ is satisfiable, then using the satisfying assignment for x and the intended values for the S_{ijk}'s we obtain an element of $\mathcal{S}$ with m non-zero entries. Note that for each constraint involving variables x_i, x_j, x_k, exactly one of the four variables $S_{ijk}(\cdot, \cdot)$ is non-zero.

On the other hand, note that if the function $S_{ijk}(\cdot, \cdot)$ has exactly one non-zero entry it must be that the induced values of (x_i, x_j, x_k) satisfy the constraint $x_k = 1 + x_i \cdot x_j$ (which one can see either by trying all such S_{ijk} or noting that each of the four different satisfying assignments to (x_i, x_j, x_k) gives a unique such S_{ijk}). Since every S_{ijk} is constrained to have an odd number of non-zero entries by Equation (1), it means that whenever S_{ijk} induces values of (x_i, x_j, x_k) that do not satisfy $x_k = 1 + x_i \cdot x_j$, it must hold that S_{ijk} has three non-zero entries. Therefore, we see that if $\mathsf{Opt}(\Psi) \leq 1 - \delta$, it must hold that every element of $\mathcal{S}$ has at least $(1 + 2\delta)m$ non-zero entries.

To summarize, we obtain that it is NP-hard to approximate the minimum weight element of an affine subspace (or equivalently, the Nearest Codeword Problem) to within a constant factor $1 + 2\delta$.

3.2 Reduction to Minimum Distance

To get the hardness result for the Min Dist problem, we would like to alter the reduction in the previous section so that it produces a linear subspace rather than an affine one. The only non-homogenous part of the subspace produced are the equations (1) constraining each S_{ijk} to have an odd number of entries. To produce a linear subspace, we are going to replace the constant 1 with a variable x_0, which is intended to take the value 1. In other words, we replace Equation (1) with the following equation:

$$S_{ijk}(0,0) + S_{ijk}(0,1) + S_{ijk}(1,0) + S_{ijk}(1,1) = x_0 \tag{1'}$$

However, in order to make this work we need to ensure that every assignment where x_0 is set to 0 has large weight, and this requires adding some more components to the reduction.

A first observation is that the system of constraints relating S_{ijk} to (x_0, x_i, x_j, x_k) is invertible. Namely, we have Equations (1')-(4), and inversely, that

$$\begin{aligned} S_{ijk}(0,0) &= x_i + x_j + x_k & \qquad S_{ijk}(0,1) &= x_0 + x_j + x_k \\ S_{ijk}(1,0) &= x_0 + x_i + x_k & \qquad S_{ijk}(1,1) &= x_0 + x_k. \end{aligned}$$

Second, if $x_0 = 0$ but at least one of (x_i, x_j, x_k) is non-zero, it must hold that S_{ijk} has at least two non-zero entries. Thus, if it happens that for a large fraction (more than $1/2$) of constraints at least one of (x_i, x_j, x_k) is non-zero, it must be the case that the total weight of the S_{ijk}'s is larger than m. But of course, we have no way to guarantee such a condition on (x_i, x_j, x_k).

However, we can construct what morally amounts to a separate dummy instance of Max NAND that has this property, and then let it use the same x_0 variable as Ψ. Towards this end, let $C \subseteq \mathbb{F}_2^N$ be a linear code of relative distance $1/2 - \epsilon$. Here $\epsilon > 0$ will be chosen sufficiently small and for reasons that will become clear momentarily, the dimension of the code will be exactly n so that one can take $N = O(n)$.

Now we introduce $N + N^2$ new variables which we think of as a vector $y \in \mathbb{F}_2^N$ and matrix $Y \in \mathbb{F}_2^{N \times N}$. The vector y should be an element of C and the matrix Y should be an element of $C \otimes C$. The intention is that $Y = y \cdot y^\top$, or in other words, that for every $i, j \in [N]$ we have $Y_{ij} = y_i \cdot y_j$.

Analogously to the S_{ijk} functions intended to check the NAND constraints of Ψ, we now introduce for every $i, j \in [N]$ a function $Z_{ij} : \mathbb{F}_2^2 \to \mathbb{F}_2$ that is intended to check the constraint $Y_{ij} = y_i \cdot y_j$, and that is supposed to be the indicator of the assignment to the variables (y_i, y_j). We then impose the analogues of the constraints (1')-(4), viz.

$$\begin{aligned} Z_{ij}(0,0) + Z_{ij}(0,1) + Z_{ij}(1,0) + Z_{ij}(1,1) &= x_0 && (5) \\ Z_{ij}(1,0) + Z_{ij}(1,1) &= y_i && (6) \\ Z_{ij}(0,1) + Z_{ij}(1,1) &= y_j && (7) \\ Z_{ij}(1,1) &= Y_{ij}. && (8) \end{aligned}$$

Figure 1 gives an overview of the different components of the reduction and their relations (including some relations that we have not yet described, though we shall do so momentarily).

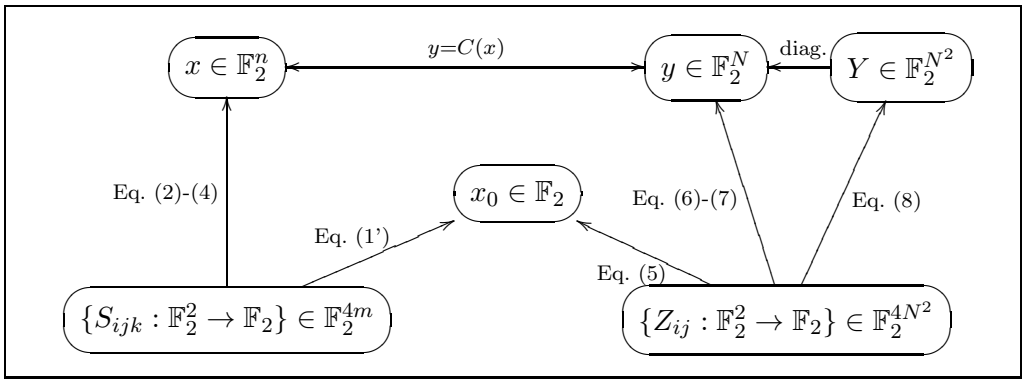

Figure 1: The different components of the reduction to MIN DIST(2). An arrow from one component to another indicates that the second component is a linear function of the first, with the label indicating the nature of this linear function.

The final subspace $\mathcal{S}$ will consist of the projection to the $4m$ different S_{ijk} variables and the $4N^2$ different Z_{ij} variables, but with each of the S_{ijk} variables repeated some $r \approx N^2/m$ number of times in order to make these two sets of variables of comparable size.

Note that by Equations (1)-(4) and (5)-(8) these variables uniquely determine x_0, x, y and Y. Furthermore, because of the invertibility of these constraints, we have that if some S_{ijk} or Z_{ij} is non-zero it must hold that one of x_0, x, y and Y are non-zero.

As in the previous section, when x_0 is non-zero, each S_{ijk} and Z_{ij} must have at least one non-zero entry and all the δ fraction of the S_{ijk}'s corresponding to unsatisfied NAND constraints of Ψ must have at least three non-zero entries, giving a total weight of

$$(1 + 2\delta)rm + N^2.$$

Now consider the case that x_0 is zero. Let us first look at the subcase that y is non-zero. Since $y \in C$ is a non-zero codeword, at least $(1/2 - \epsilon)N$ of its coordinates are non-zero. Thus, for at least $(3/4 - 2\epsilon)N^2$ pairs $(y_i, y_j) \neq (0, 0)$. For each such pair, the corresponding Z_{ij} function is non-zero, and as argued earlier, has at least two non-zero entries, which means that the total weight of the Z_{ij}'s is at least

$$2 \cdot (3/4 - 2\epsilon) \cdot N^2 = \left(\frac{3}{2} - 4\epsilon\right) \cdot N^2.$$

The next subcase is that x_0 and y are zero but either x or Y is non-zero. We first enforce that $x = 0$. Recall that C has dimension exactly n, and hence there is a one-to-one linear map $C : \mathbb{F}_2^n \mapsto \mathbb{F}_2^N$. We may therefore add the additional constraints that $y = C(x)$ is the encoding of x. Then, x is non-zero if and only if y is.

The only possibility that remains is that x_0, x and y are all zero, but that the matrix Y is non-zero. In this case, it is easily verified from Equations (5)-(8) that for each $i, j \in [N]$ such that Y_{ij} is non-zero, it must be that Z_{ij} has four non-zero entries. However, the distance of the code $C \otimes C$ to which Y belongs is only $(1/2 - \epsilon)^2 < 1/4$, so it seems as though we just came short of obtaining a large distance. However, there are two additional constraints

that we can impose on Y: first, if $Y = y \cdot y^\top$ we have that the diagonal entries Y_{ii} should equal $y_i^2 = y_i$, so we can add the requirement that the diagonal of Y equals y. Second, it should be the case that $Y_{ij} = Y_{ji}$, so we also add the constraint that Y is symmetric. With these constraints, Lemma 2.6 now implies that Y in fact has $(1/2 - \epsilon)^2 \cdot \frac{3}{2} > (1/4 - 2\epsilon)\frac{3}{2}$ fraction non-zero entries. As mentioned above, each corresponding Z_{ij} function has four non-zero entries giving a total of

$$4 \cdot (3/8 - 3\epsilon) \cdot N^2 = \left(\frac{3}{2} - 12\epsilon\right) \cdot N^2$$

non-zero entries.

In summary, this gives that when $\mathsf{Opt}(\Psi) \leq 1 - \delta$, every non-zero vector in $\mathcal{S}$ must have weight at least

$$\min\left((1 + 2\delta)rm + N^2,\ \left(\frac{3}{2} - 12\epsilon\right) \cdot N^2\right),$$

whereas if Ψ is satisfiable the minimum distance is $rm + N^2$ (since exactly one entry is non-zero for each S_{ijk} and Z_{ij}). Choosing $\epsilon > 0$ sufficiently small and

$$r \approx \frac{N^2}{2(1 + 2\delta)m}$$

we obtain that it is NP-hard to approximate Min Dist(2) to within some factor $\delta' > 1$.

We have not yet proved that $\mathcal{C}(\Psi)$ has good rate and distance. In Section 5.3, we give a proof of this for our reduction for the general case. That proof also works for the binary case.

4 Interlude: Polynomials and Pseudorandomness over $\mathbb{F}_q$

In this section we describe some background material that we need for the generalization of the reduction for $\mathbb{F}_2$ to any finite field.

We recall two basic properties about polynomials over finite fields. First, we have the well-known fact that every function on $\mathbb{F}_q^n$ can be uniquely represented by a polynomial of maximum degree $q - 1$.

Fact 4.1. The set of polynomials

$$\{\, X_1^{i_1} X_2^{i_2} \cdots X_n^{i_n} \ :\ 0 \leq i_j \leq q - 1 \text{ for all } 1 \leq j \leq n \,\}$$

form a basis for the set of functions from $\mathbb{F}_q^n$ to $\mathbb{F}_q$.

Second, we have the Schwarz-Zippel Lemma.

Lemma 4.2 (Schwarz-Zippel)**.** *Let $p \in \mathbb{F}_q[X_1, \ldots, X_n]$ be a non-zero polynomial of total degree at most d. Then p has at most a fraction dq^{n-1} zeros.*

4.1 Linear Approximations to Nonlinear Codes

In our hardness result for MIN DIST(q), we need explicit constructions of certain codes which can be thought of as serving as linear approximations to some nonlinear codes. In particular, we need a sequence of linear codes $C_1, \ldots, C_{q-1}$ over $\mathbb{F}_q^N$ with the following two properties:

1. $d(C_e) \gtrsim (1 - e/q) \cdot N$ for $1 \leq e \leq q-1$.
2. If $x \in C_1$ then $x^e \in C_e$ for $1 \leq e \leq q-1$. Here x^e denotes a vector that is componentwise e^{th} power of x.

In other words, C_e should contain the nonlinear code $\{x^e\}_{x \in C_1}$, while still having a reasonable amount of distance. In this sense we can think of C_e as a linear approximation to a nonlinear code.

To obtain such a sequence of codes, we use pseudorandom generators for low-degree polynomials. Such pseudorandom generators were recently constructed by Viola [16] (building on [4, 13]), who showed that the sum of d PRGs for linear functions fool degree d polynomials. Using his result, and PRGs against linear functions of optimal seed length $\log_q n + O(1 + \log_q 1/\epsilon)$ (see e.g., Appendix A of [4]), one obtains the following theorem.

Theorem 4.3. *For every prime power q, $d > 0$, $\epsilon > 0$ there is a constant $c := c(q, d, \epsilon)$ such that for every $n > 0$, there is a polynomial time constructible (multi)set $R \subseteq \mathbb{F}_q^n$ of size $|R| \leq c \cdot n^d$ such that, for any polynomial $f : \mathbb{F}_q^n \to \mathbb{F}_q$ of total degree at most d, it holds that*

$$\sum_{a \in \mathbb{F}_q} \left| \Pr_{x \sim R}[f(x) = a] - \Pr_{x \sim \mathbb{F}_q^n}[f(x) = a] \right| \leq \epsilon. \tag{9}$$

Remark 4.4. The constant c of Theorem 4.3 can be taken to be $c(q, d, \epsilon) = (q/\epsilon)^{O(d2^d)}$.

Remark 4.5. In order for the hardness result of Theorem 1.1 to apply for codes with constant rate, we need the set R of Theorem 4.3 to have size $O(n^d)$. For this, the parameters of Viola's result [16] are necessary, and the earlier result [13] does not suffice. If one does not care about this property, any $|R| = \text{poly}(n)$ suffices.

A simple corollary of the property (9) and the Schwarz-Zippel Lemma 4.2 is the following.

Corollary 4.6. *If $d = q - 1$ the (multi)set $R \subseteq \mathbb{F}_q^n$ constructed in Theorem 4.3 has the property that for every non-zero polynomial $f : \mathbb{F}_q^n \to \mathbb{F}_q$ of total degree at most $e \leq q - 1$,*

$$\Pr_{x \sim R}[f(x) \neq 0] \geq 1 - e/q - \epsilon. \tag{10}$$

Now define, for $1 \leq e \leq q-1$, C_e to be the set of all vectors $(f(x))_{x \in R}$ where $f : \mathbb{F}_q^n \mapsto \mathbb{F}_q$ is a degree e polynomial with no constant term (i.e., $f(0) = 0$). Clearly, C_e is a linear subspace of $\mathbb{F}_q^{|R|}$. As observed in Corollary 4.6, the relative distance of C_e is essentially $1 - e/q$ (as ϵ can be taken to be arbitrarily small relative to q). Moreover, any $v \in C_1$ is the evaluation vector of a degree one polynomial, and hence v^e is the evaluation vector of a degree e polynomial, and therefore $v^e \in C_e$ as desired.

5 Reduction to Min Dist(q) for $q \geq 3$

We now describe a general reduction from the Max NAND problem to the Min Dist(q) problem for any prime power q. The basic idea is the same as in the $\mathbb{F}_2$ case but some additional work is needed both in the reduction itself and its analysis.

Given a Max NAND instance Ψ, we construct a linear code $\mathcal{C}(\Psi)$ over $\mathbb{F}_q$ as follows. For simplicity we here assume that $q \geq 3$ as the binary case was already handled in the previous section. As before, let n be the number of variables in the Max NAND instance and m the number of constraints.

Fix some small enough parameter ϵ and let $R \subseteq \mathbb{F}_q^n$ be the ϵ-pseudorandom set for degree $q-1$ polynomials $\mathbb{F}_q^n \to \mathbb{F}_q$ given by Theorem 4.3. Let $N = |R| = O(n^{q-1})$.

For $0 \leq d \leq q-1$, let $P_d \subseteq \mathbb{F}_q^N$ be the linear subspace of all degree d polynomials in n variables with coefficients in $\mathbb{F}_q$ and no constant term, evaluated at points on R. I.e., all vectors in P_d are of the form $(p(x))_{x \in R}$ for some polynomial $p \in \mathbb{F}_q[X_1, \ldots, X_n]$ with $\deg(p) \leq d$ and $p(0) = 0$. Note that P_d is a linear code and by Corollary 4.6, its relative distance is at least $1 - d/q - \epsilon$.

We define $C = P_1$ and for $\alpha \in \mathbb{F}_q^n$ we write $C(\alpha) \in \mathbb{F}_q^N$ for the encoding of α under C; this corresponds to the evaluations of the linear polynomial $\sum_{i=1}^n \alpha_i X_i$ at all points $(X_1, \ldots, X_n)$ in R. Conversely, for a codeword $y \in C$ we write $\alpha = C^{-1}(y) \in \mathbb{F}_q^n$ for the (unique) decoding of y.

From here on, we will ignore the parameter $\epsilon > 0$; it can be chosen to be sufficiently small (independent of q and the inapproximability for Max NAND) and hence the effect of this can be made insignificant.

We now construct a linear code $\mathcal{C}'(\Psi)$ with variables as described in Figure 2. As in the $\mathbb{F}_2$ case, the final code $\mathcal{C}(\Psi)$ will consist of the projection of these variables to the Z_{ij}'s and the S_{ijk}'s, which determine the remaining variables by the constraints that we shall define momentarily.

1. For every $0 \leq e \leq 2(q-1)$ a vector $Y^e \in \mathbb{F}_q^N$.
2. For every $0 \leq e, f \leq q-1$ a matrix $Y^{e,f} \in \mathbb{F}_q^{N^2}$.
3. For every $1 \leq i, j \leq N$ a function $Z_{ij} : \mathbb{F}_q^2 \to \mathbb{F}_q$ (i.e., a vector in $\mathbb{F}_q^{q^2}$).
4. For every equation $x_k = 1 + x_i \cdot x_j$ in Ψ, a function $S_{ijk} : \mathbb{F}_2^2 \to \mathbb{F}_q$ (i.e., a vector in $\mathbb{F}_q^4$).

Figure 2: Variables of $\mathcal{C}'(\Psi)$.

Before we describe the constraints defining $\mathcal{C}'(\Psi)$ it is instructive to describe the intended values of these variables. Loosely speaking, the different Y variables are supposed to be an encoding of an assignment $\alpha \in \mathbb{F}_2^n$ to Ψ, the function S_{ijk} is a check that α satisfies the equation $x_k = 1 + x_i \cdot x_j$, and the Z_{ij} functions check that the Y variables resemble a valid encoding of some α.

Specifically, the variables are supposed to be assigned as described in Figure 3.

1. Y^e is supposed to be $C(\alpha)^e$ (where we think of $\mathbb{F}_2^n$ as a subset of $\mathbb{F}_q^n$ in the obvious way) .
2. Y^{ef} is supposed to be $C(\alpha)^e \cdot (C(\alpha)^f)^\top$ (i.e., we should have $Y^{ef}(i,j) = C(\alpha)_i^e C(\alpha)_j^f$.
3. Z_{ij} is supposed to be the indicator function of $(C(\alpha)_i, C(\alpha)_j)$ (i.e., $Z_{ij}(x,y)$ should be 1 if $x = C(\alpha)_i$ and $y = C(\alpha)_j$; and 0 otherwise).
4. S_{ijk} is supposed to be the indicator function of (α_i, α_j) (i.e., $S_{ijk}(a,b) = 1$ if $\alpha_i = a$ and $\alpha_j = b$; and 0 otherwise).

Figure 3: Intent of variables of $\mathcal{C}'(\Psi)$.

We categorize the constraints of $\mathcal{C}'(\Psi)$ as being of two different types, namely *basic constraints* that aim to enforce rudimentary checks of Items 1 and 2 of Figure 3, and *consistency constraints* that aim to use the Z_{ij}'s and S_{ijk}'s to check that the Y^{ef} matrices are consistent with an encoding of a good assignment to Ψ. As a comparison with the reduction for $\mathbb{F}_2$ in Section 3, the basic constraints correspond to the horizontal arrows on the upper side of Figure 1, and the consistency constraints correspond to the other arrows, i.e., Equations (1')-(8).

Keeping the interpretation from Figure 3 in mind, the basic constraints that we impose are given in Figure 4.

1. For $0 \leq e \leq q-1$, $Y^e \in P_e$.
2. For $q \leq e \leq 2(q-1)$, $Y^e = Y^{e-(q-1)}$.
3. For $0 \leq e, f \leq q-1$:
   (a) $Y^{ef} \in P_e \otimes P_f$.
   (b) The diagonal of Y^{ef} equals Y^{e+f}.
4. For $0 \leq e \leq q-1$, the rows (resp. columns) of $Y^{0,e}$ (resp. $Y^{e,0}$) are identical (and therefore equal to Y^e as this is the diagonal).
5. The matrix $Y^{q-1,q-1}$ is symmetric[1].

Figure 4: Basic constraints of $\mathcal{C}'(\Psi)$.

Note that all entries of the matrix $Y^{0,0}$ must be equal, and that in the intended assignment they should equal the constant 1. For notational convenience let us write $Y_0 \in \mathbb{F}_q$ for the value of the entries of $Y^{0,0}$ (this variable plays the same role as the variable x_0 in the reduction for $\mathbb{F}_2$ in Section 3).

[1]In general we could add the constraint that $Y^{e,f} = (Y^{f,e})^\top$ for every e, f, but it turns out we only need it for the case $e = f = q-1$.

We then turn to the consistency constraints of $\mathcal{C}'(\Psi)$, which are described in Figure 5.

1. For every constraint $x_k = 1 + x_i \cdot x_j$ of Ψ, four constraints on S_{ijk}:

$$\begin{aligned} Y_0 &= \sum_{a,b\in\mathbb{F}_2} S_{ijk}(a,b) \qquad \alpha_i = \sum_{a,b\in\mathbb{F}_2} a \cdot S_{ijk}(a,b) \\ \alpha_j &= \sum_{a,b\in\mathbb{F}_2} b \cdot S_{ijk}(a,b) \quad \alpha_k = \sum_{a,b\in\mathbb{F}_2} (1 \oplus a \cdot b) \cdot S_{ijk}(a,b). \end{aligned} \tag{11}$$

   (Here $\oplus$ denotes addition in $\mathbb{F}_2$ and the remaining summations are over $\mathbb{F}_q$.)

2. For every $i, j \in [N]$, q^2 constraints on Z_{ij}: for every $0 \leq e, f \leq q-1$ it must hold that

$$Y^{e,f}(i,j) \quad = \quad \sum_{x,y\in\mathbb{F}_q} x^e y^f Z_{ij}(x,y). \tag{12}$$

Figure 5: Consistency constraints of $\mathcal{C}'(\Psi)$.

The four equations (11) are the same as Equations (1)-(4) from the $\mathbb{F}_2$ reduction, the only difference being that they are now constraints over $\mathbb{F}_q$. Note that instead of Y_0 we would like to use the constant 1 in the above constraint, but as we are not allowed to do this we use Y_0, which, as mentioned above, is intended to equal 1. Note also that $Y^1 = C(\alpha)$, and thus α is implicitly defined by Y^1. If one wanted to be precise, one would write $C^{-1}(Y^1)_i$ instead of α_i in the above equations.

Note that the function S_{ijk} is an invertible linear transformation of $\{Y_0, \alpha_i, \alpha_j, \alpha_k\}$ and hence is non-zero if and only if one of those four variables are non-zero. Similarly, from (12) it follows that Z_{ij} is an invertible linear transformation of the set of (i,j)'th entries of the q^2 different matrices $\{Y^{ef}\}_{0\leq e,f\leq q-1}$ (this is an immediate consequence of Fact 4.1). In particular Z_{ij} is non-zero if and only if the (i,j)'th entry of some matrix $Y^{e,f}$ is non-zero.

The final code $\mathcal{C}(\Psi)$ contains the projection of these variables to the functions Z_{ij} and the functions S_{ijk}, with each S_{ijk} repeated $r \geq 1$ times. Note that $\mathcal{C}(\Psi)$ is a subspace of $\mathbb{F}_q^M$ where $M = (qN)^2 + 4rm$. The completeness and soundness are as follows.

Lemma 5.1 (Completeness)**.** *If* $\mathsf{Opt}(\Psi) = 1$ *then*

$$d(\mathcal{C}(\Psi)) \leq N^2 + rm.$$

Lemma 5.2 (Soundness)**.** *If* $\mathsf{Opt}(\Psi) \leq 1 - \delta$ *then*

$$d(\mathcal{C}(\Psi)) \geq \min\left(N^2 + (1+\delta)rm, (1+1/q)N^2\right).$$

Lemma 5.3 ($\mathcal{C}$ is a Good Code)**.** *The dimension of* $\mathcal{C}(\Psi)$ *is* $\Omega(N^2)$*, and the distance is at least* N^2*.*

Setting $r \approx \frac{N^2}{(1+\delta)qm}$, Lemmas 5.1-5.3 give Theorem 1.1 (for the case $q \geq 3$).
In the following three subsections we prove the three lemmas.

5.1 Proof of Completeness

We first consider the Completeness Lemma 5.1, which is straightforward to prove.

Proof of Lemma 5.1. Given a satisfying assignment $\alpha \in \mathbb{F}_2^n$ to the set of quadratic equations, we construct a good codeword by following the intent described in Figure 3. Clearly this satisfies all the basic constraints.

To check the constraints on Z_{ij}, recall that it is defined as

$$Z_{ij}(x,y) = \begin{cases} 1 & \text{if } (x,y) = (C(\alpha)_i, C(\alpha)_j) \\ 0 & \text{otherwise.} \end{cases} .$$

This choice of Z_{ij} satisfies its q^2 constraints since for any $0 \leq e, f \leq q-1$

$$\sum_{x,y} x^e y^f Z_{ij}(x,y) = C(\alpha)_i^e C(\alpha)_j^f = Y^{ef}(i,j).$$

Analogously, for the constraints on S_{ijk} we have

$$S_{ijk}(a,b) = \begin{cases} 1 & \text{if } (a,b) = (\alpha_i, \alpha_j) \\ 0 & \text{otherwise.} \end{cases} ,$$

which is again easily verified to satisfy its four constraints and hence this constitutes a codeword.

The weight of the codeword is $N^2 + rm$, since each Z_{ij} and each S_{ijk} has exactly one non-zero coordinate. □

5.2 Proof of Soundness

In this section we prove the Soundness Lemma 5.2, which is the part that requires the most work. Let us first describe the intuition.

In the analysis, we view codewords where $Y_0 \neq 0$ as resembling a valid encoding of some $\alpha \in \mathbb{F}_2^n$ and for these we shall argue that small weight corresponds to a good assignment to Ψ.

Most of the complication comes from analysing codewords where $Y_0 = 0$, which we think of as not resembling a valid encoding of some α. For such codewords we argue that there must be a lot of weight on the Z_{ij}'s. To pull off this argument, we look at a non-zero $Y^{e,f}$ that has $d = e + f$ minimal. Then we look at the set of Z_{ij}'s that are non-zero. The total number of such Z_{ij}'s can be lower bounded using the distance bound on $Y^{e,f}$ (though this bound unfortunately gets worse as d increases). The fact that every $Y^{e',f'}$ with $e' + f' < d$ is zero gives a set of $\Theta(d^2)$ linear constraints on every such Z_{ij}. These constraints induce a linear code over $\mathbb{F}_q^{q^2}$ to which each Z_{ij} must belong. We then argue that as d increases, the distance of this linear code increases as well, meaning that the non-zero Z_{ij}'s must have an increasingly larger number of non-zero entries. This increased distance balances the decrease in the number of non-zero Z_{ij}'s, allowing us to conclude that no matter the value of d, the total number of non-zero entries among all the Z_{ij}'s is always large.

Before we proceed with the formal proof of the soundness, let us state two lemmas that we use to obtain lower bounds on the distance of Z_{ij}. The proofs of these two lemmas can be found in Section 6. First, we have a lemma for the case when d is small.

Lemma 5.4. *Suppose $f : \mathbb{F}_q \times \mathbb{F}_q \to \mathbb{F}_q$ is a non-zero function satisfying*

$$\sum_{x,y \in \mathbb{F}_q} x^a y^b f(x,y) = 0$$

for every (a,b) such that $0 \le a,b \le q-1$ and $a+b < d$ for some $0 \le d \le q-1$. Then $f(x,y) \neq 0$ for at least $d+1$ points in $\mathbb{F}_q^2$.

Second, we have a lemma for the case when d is large.

Lemma 5.5. *Suppose $f : \mathbb{F}_q \times \mathbb{F}_q \to \mathbb{F}_q$ is a non-zero function satisfying*

$$\sum_{x,y \in \mathbb{F}_q} x^a y^b f(x,y) = 0$$

for every (a,b) such that $0 \le a,b \le q-1$ and $a+b < d$ for some $q-1 \le d \le 2(q-1)$. Then $f(x,y) \neq 0$ for at least $q(d+2-q)$ points in $\mathbb{F}_q^2$.

We are now ready to proceed with the proof of soundness.

Proof of Lemma 5.2. Let $\{Z_{ij}\}_{i,j\in[N]}$ and $\{S_{ijk}\}_{(i,j,k)\in\Psi}$ be some non-zero codeword of $\mathcal{C}(\Psi)$, and consider the induced values of the Y variables.

Let (e,f) be such that $Y^{e,f}$ is non-zero and $e+f$ is minimal (breaking ties arbitrarily). Since the codeword is non-zero it follows that such an (e,f) exists (by invertibility of (11) and (12)).

We do a case analysis based on the value of $e+f$.

Case 1: $e = f = 0$. This is the case when $Y_0 \neq 0$. In other words, we think of the Y variables as resembling a valid encoding of some assignment to Ψ, so that the soundness of Ψ comes into play.

If $e = f = 0$ we have that all Z_{ij}'s and S_{ijk}'s are non-zero and hence the weight is at least $N^2 + rm$. We will show that the soundness condition of Ψ implies that a δ fraction of the S_{ijk}'s must in fact have two non-zero entries, so that the total weight of the codeword is at least

$$N^2 + (1+\delta)rm.$$

To see this, construct an assignment to the quadratic equations instance as follows. Let $\alpha = C^{-1}(Y) \in \mathbb{F}_q^n$. From the α_i, $i \in [n]$, we define a boolean assignment β_i as follows: $\beta_i = 0$ if $\alpha_i = 0$, and $\beta_i = 1$ otherwise. We claim that every constraint $x_k = 1 + x_i \cdot x_j$ for which S_{ijk} only has a single non-zero entry is satisfied by β. Indeed, suppose that $S_{ijk}(a,b) = c \neq 0$ and all other values of S_{ijk} are 0. Then the constraints on S_{ijk} imply that

$$\alpha_i = a \cdot c \qquad \alpha_j = b \cdot c \qquad \alpha_k = (1 \oplus ab) \cdot c.$$

which implies that $\beta_i = a$, $\beta_j = b$, and $\beta_k = 1 \oplus ab = 1 \oplus \beta_i \cdot \beta_j$. By the soundness assumption $\mathsf{Opt}(\Psi) \le 1-\delta$, and hence at least a δ fraction of the constraints are not satisfied by β; the corresponding S_{ijk}'s must therefore have at least two non-zero entries.

Case 2: $0 < e+f < q-1$**.** Let $d = e+f$. The minimality of $e+f$ implies that $Y^{a,b} \equiv 0$ for all $a+b < d$. From Equation (12), we have that for all $a+b < d$, $\sum_{x,y\in\mathbb{F}_q} x^a y^b Z_{ij}(x,y) = 0$. Applying Lemma 5.4, each non-zero Z_{ij} has at least $d+1$ non-zero entries. Furthermore the fraction of non-zero Z_{ij}'s is at least $1-d/q$. This is because the distance of the codes P_e and P_f is at least $1-e/q$ and $1-f/q$ respectively, and hence the distance of the code $P_e \otimes P_f$ is at least $(1-e/q)(1-f/q) \geq 1-d/q$. Thus at least a $1-d/q$ fraction of entries of $Y^{e,f}$ are non-zero and by Equation (12), the same applies to Z_{ij}. Hence the total number of non-zero entries over all $Z_{ij}(\cdot,\cdot)$ is at least

$$N^2(1-d/q)(d+1) \geq N^2 \frac{2(q-1)}{q} \geq \frac{4}{3}N^2,$$

where the first inequality follows by noting that for $1 \leq d \leq q-2$ the left hand side is minimized by $d=1$ and $d=q-2$, and the second inequality follows from the assumption $q \geq 3$.

Case 3: $e+f = q-1$**.** In this case, either of Lemma 5.4 or Lemma 5.5 gives that any non-zero Z_{ij} has q non-zero entries.

The fraction of Z_{ij}'s that are non-zero is at least $(1-e/q)(1-f/q) = 1/q + ef/q^2$. Unfortunately, if $ef = 0$ this bound is not good enough. However, note that if $Y^{0,q-1}$ (or $Y^{q-1,0}$) is non-zero then so is Y^{q-1} (by Figure 4, item 4) implying that $Y^{q-2,1}$ is non-zero (since by Figure 4, item 3(b), it has Y^{q-1} as diagonal). Hence we may assume without loss of generality that $ef \geq q-2$ so that at least a fraction $1/q + (q-2)/q^2 = 2(q-1)/q^2$ of the Z_{ij}'s are non-zero.

Thus we see that the total weight of the codeword is at least

$$N^2 \cdot \frac{2(q-1)}{q^2} \cdot q = N^2 \cdot \frac{2(q-1)}{q} \geq \frac{4}{3}N^2.$$

Case 4: $q-1 < e+f < 2(q-1)$**.** Let $e+f = q-1+s$ for $1 \leq s < q-1$. In this case, Lemma 5.5 gives that any non-zero Z_ij has $q \cdot (e+f+2-q) = q(s+1)$ non-zero entries. The fraction of Z_{ij}'s that are non-zero is at least $(1-e/q)(1-f/q) = 1-(e+f)/q+ef/q^2$. Furthermore, since $0 \leq e, f \leq q-1$ we must have that $\min(e,f) \geq s$ so that $ef \geq s(q-1)$. Hence

$$1-(e+f)/q+ef/q^2 \geq 1 - \frac{q-1+s}{q} + \frac{s(q-1)}{q^2} = \frac{q-s}{q^2}$$

Thus, the total weight of all the Z_{ij}'s is lower bounded by

$$N^2 \cdot \frac{q-s}{q^2} \cdot q(s+1) = N^2 \cdot \frac{(q-s)(s+1)}{q} \geq N^2 \cdot \frac{2(q-1)}{q} \geq \frac{4}{3}N^2.$$

Case 5: $e+f = 2(q-1)$**.** The only remaining case is when $e = f = q-1$. Now Lemma 5.5 gives that any non-zero Z_{ij} has q^2 non-zero entries. On the other hand, 'a priori, the distance of $Y^{q-1,q-1}$ is as small as $1/q^2$, which seems problematic. However, we still have some leeway: recall that the diagonal of $Y^{q-1,q-1}$ should equal $Y^{2(q-1)} = Y^{q-1}$ which also happens to be the diagonal of $Y^{q-1,0}$ (Figure 4, items 3(b) and 2). Since $Y^{q-1,0}$

is identically 0 this means that the diagonal of $Y^{q-1,q-1}$ has to be zero. By Lemma 2.6, we can then conclude that at least a fraction $\frac{1}{q^2} \cdot (1 + 1/q)$ of the Z_{ij}'s are non-zero. As each such Z_{ij} has q^2 non-zero entries, we see that the total weight of the codeword is at least

$$N^2 \cdot (1 + 1/q).$$

This concludes the proof of Lemma 5.2. □

5.3 Proof That The Code Is Good

In this section we prove Lemma 5.3, that $\mathcal{C}(\Psi)$ is a good code. After the soundness analysis, this becomes relatively easy. To get the bound on the rate of the code, we need the following simple lower bound on the rate of a certain restricted tensor product of a code.

Claim 5.6. *Let $C \subseteq \mathbb{F}_q^n$ be a linear code and $\tilde{C}$ be the linear subspace of $C \otimes C$ where every codeword is restricted to be symmetric. Then* $\dim(\tilde{C}) \geq \dim(C)^2/2$.

Proof. Let $G \in \mathbb{F}_q^{n \times k}$ be the generator matrix of C, where $k = \dim(C)$. It is easy to check that the generator matrix of $C \otimes C$ is $G \otimes G \in \mathbb{F}_q^{n^2 \times k^2}$. We think of $G \otimes G$ as mapping a $k \times k$ matrix X to an $n \times n$ matrix $Y = (G \otimes G)X$ where

$$Y_{i_1,i_2} = \sum_{j_1,j_2 \in [k]} g_{i_1,j_1} g_{i_2,j_2} X_{j_1,j_2}.$$

It is easily verified that if X is symmetric then so is Y, so the dimension of $\tilde{C}$ is at least the dimension of the space of symmetric $k \times k$ matrices over $\mathbb{F}_q$, which equals $\frac{k(k+1)}{2} \geq k^2/2$ □

We can now prove that $\mathcal{C}(\Psi)$ is a good code.

Proof of Lemma 5.3. Let us first consider the distance of $\mathcal{C}(\Psi)$. In Lemma 5.2, it is shown that any codeword for which $Y_0 = 0$ has at least $N^2(1 + 1/q) \geq N^2$ non-zero entries. On the other hand, if $Y_0 \neq 0$ each Z_{ij} and S_{ijk} must have at least one non-zero entry, for a total of $N^2 + rm \geq N^2$ non-zero entries.

It remains to prove that $\mathcal{C}(\Psi)$ has large dimension, which requires a little more work. Let $\alpha \in \mathbb{F}_q^n$ and assign every matrix $Y^{e,f}$ except $Y^{q-1,q-1}$ according to the intent of Figure 3. I.e., for $(e, f) \neq (q-1, q-1)$ we set $Y^{ef}(i, j) = C(\alpha)_i^e C(\alpha)_j^f$.

We shall show that there are still $q^{\Omega(N^2)}$ ways to choose $Y^{q-1,q-1}$ so that the resulting set of values satisfy the basic constraints of Figure 4. Then, from the invertibility of Equations (11) and (12) of Figure 5, it follows that each of these $q^{\Omega(N^2)}$ ways to choose $Y^{q-1,q-1}$ extends to a unique codeword of $\mathcal{C}(\Psi)$.

By Claim 5.6, the space of matrices $Y^{q-1,q-1}$ satisfying Items 3(a) and 5 of Figure 4 has dimension at least $\dim(P_{q-1})^2/2 \geq n^{2(q-1)}/2 = \Omega(N^2)$ (recall that $N = O(n^{q-1})$). The only additional constraint on $Y^{q-1,q-1}$ is Item 3(b) of Figure 4, that the diagonal has to be $Y^{2(q-1)} = Y^{q-1}$. However, this can reduce the dimension by at most N, so the remaining dimension is still $\Omega(N^2)$.

□

6 Combinatorial Lemmas

In this section we prove the combinatorial lemmas used in the proof of Lemma 5.2.

Lemma 5.4 restated. Suppose $f : \mathbb{F}_q \times \mathbb{F}_q \to \mathbb{F}_q$ is a non-zero function satisfying

$$\sum_{x,y \in \mathbb{F}_q} x^a y^b f(x,y) = 0$$

for every (a,b) such that $0 \le a, b \le q-1$ and $a+b < d$ for some $0 \le d \le q-1$. Then $f(x,y) \neq 0$ for at least $d+1$ points in $\mathbb{F}_q^2$.

Proof. Let $X = \{\, x \,:\, \exists y\, f(x,y) \neq 0 \,\}$ and $Y = \{\, y \,:\, \exists x\, f(x,y) \neq 0 \,\}$. Without loss of generality, assume that $|X| \ge |Y|$. Define $g : \mathbb{F}_q \to \mathbb{F}_q$ by

$$g(x) = \sum_{y \in \mathbb{F}_q} f(x,y).$$

First suppose g is non-zero. Then we use the fact that

$$\sum_x x^a g(x) = \sum_{x,y} x^a y^0 f(x,y) = 0$$

for every $a < d$, which implies that g has to be non-zero in at least $d+1$ points. This is because in the $d \times q$ matrix whose rows are $(x^a)_{x \in \mathbb{F}_q}$ for $0 \le a \le d-1$, any d columns form a Vandermonde matrix and hence are linearly independent. We used here the fact that $d \le q-1$. Thus f also has to be non-zero in $d+1$ points and we are done. Hence we can now assume that g is identically 0.

Let $|X| = s$ and $|Y| = t$. Since g is identically 0, it must hold that for any $x \in X$ there are at least two different y's such that $f(x,y) \neq 0$, implying that f is non-zero for at least $2s$ different points. We now show that $s + t \ge d+2$ which implies that $s \ge \frac{d+2}{2}$ (since we assumed $s \ge t$) so that f must be non-zero on at least $d+2$ points.

Consider the Vandermonde matrices

$$A_X = \begin{pmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{s-1} \\ 1 & x_2 & x_2^2 & \dots & x_2^{s-1} \\ & \vdots & & \vdots & \\ 1 & x_s & x_s^2 & \dots & x_s^{s-1} \end{pmatrix} \qquad A_Y = \begin{pmatrix} 1 & y_1 & y_1^2 & \dots & y_1^{t-1} \\ 1 & y_2 & y_2^2 & \dots & y_2^{t-1} \\ & \vdots & & \vdots & \\ 1 & y_t & y_t^2 & \dots & y_t^{t-1} \end{pmatrix},$$

where $x_1, \dots, x_s$ are the elements of X and $y_1, \dots, y_t$ are the elements of Y. Since A_X and A_Y are non-singular, so is $B := (A_X \otimes A_Y)^\top$. The matrix B is an $st \times st$ matrix such that for any $0 \le a < s$, $0 \le b < t$, its (a,b)'th row is $(x_i^a y_j^b)_{i \in [s], j \in [t]}$.

Since f is not identically zero on $X \times Y$ and B is non-singular, the dot product of f restricted to $X \times Y$ with some row of B is non-zero, i.e., there exists a row (a,b) such that

$$0 \neq \sum_{i \in [s], j \in [t]} f(x_i, y_i) x_i^a y_j^b = \sum_{x,y \in \mathbb{F}_q} x^a y^b f(x,y),$$

where for the second equality we noted that f is zero outside of $X \times Y$. From the hypothesis of the Lemma, we must have $a + b \ge d$ and therefore $s + t \ge a + b + 2 \ge d + 2$. ∎

For the next lemma we first have the following easy claim.

Claim 6.1. *Let $0 \le a \le q-2$. Then $\sum_{x\in\mathbb{F}_q} x^a = 0$.*

Proof. The case $a = 0$ is trivial. For $a > 0$, let g be a generator for $\mathbb{F}_q$ and define $h = g^a$. Since $1 \le a \le q-2$ we have $h \neq 1$ and by Fermat's little theorem we have $h^{q-1} = 1$. Thus we have

$$\sum_{x\in\mathbb{F}_q} x^a = \sum_{i=0}^{q-2} (g^i)^a = \sum_{i=0}^{q-2} h^i = \frac{h^{q-1}-1}{h-1} = 0.$$

□

Now we prove the second lemma.

Lemma 5.5 restated. Suppose $f : \mathbb{F}_q \times \mathbb{F}_q \to \mathbb{F}_q$ is a non-zero function satisfying

$$\sum_{x,y\in\mathbb{F}_q} x^a y^b f(x,y) = 0 \tag{13}$$

for every (a,b) such that $0 \le a,b \le q-1$ and $a+b < d$ for some $q-1 \le d \le 2(q-1)$. Then $f(x,y) \neq 0$ for at least $q(d+2-q)$ points in $\mathbb{F}_q^2$.

Proof. Let

$$\begin{aligned} S &= \{\,(a,b) \,:\, 0 \le a,b \le q-1,\ a+b < d\,\} \\ T &= \{\,(e,\ell) \,:\, 0 \le e,\ell \le q-1,\ e+\ell \le 2(q-1)-d\,\} \end{aligned}$$

Note that $|S| + |T| = q^2$ since the mapping $(e,\ell) \mapsto (q-1-e, q-1-\ell)$ forms a bijection from T to $\{0,1,\ldots,q-1\}^2 \setminus S$.

Now, the functions f satisfying (13) for every $(a,b) \in S$ form a linear subspace V of $\mathbb{F}_q^{q^2}$ of dimension $q^2 - |S| = |T|$.

We identify the following basis for V: for every $(e,\ell) \in T$, let $g_{e\ell}(x,y) = x^e y^\ell$. It is clear that the $g_{e\ell}$'s are linearly independent (since they are a subset of the standard polynomial basis for functions $\mathbb{F}_q^2 \to \mathbb{F}_q$; Fact 4.1) and that $|\{g_{e\ell}\}| = |T| = \dim V$, so we only have to check that each $g_{e\ell}$ indeed lies in V. We have

$$\sum_{x,y} x^a y^b g_{e\ell}(x,y) = \sum_{x,y} x^{a+e} y^{b+\ell} = \left(\sum_x x^{a+e}\right) \cdot \left(\sum_y y^{b+\ell}\right).$$

By Claim 6.1, we see that this vanishes if either $a+e < q-1$ or $b+\ell < q-1$. But this must hold, since otherwise we would have $(a+b)+(e+\ell) \ge 2(q-1)$ contradicting that $(a,b) \in S$ and $(e,\ell) \in T$.

From this we can conclude that any function $f : \mathbb{F}_q^2 \to \mathbb{F}_q$ satisfying condition (13) can be written as a polynomial of total degree at most $2(q-1)-d$. By the Schwarz-Zippel Lemma 4.2 a non-zero such f can be zero on at most a fraction $\frac{2(q-1)-d}{q}$ points of $\mathbb{F}_q^2$ and so f has to be non-zero on at least

$$q^2\left(1 - \frac{2(q-1)-d}{q}\right) = q(d+2-q)$$

points. □

References

[1] M. Ajtai. The shortest vector problem in L_2 is NP-hard for randomized reductions. In *Proc. 30th ACM Symposium on the Theory of Computing*, pages 10–19, 1998. 3

[2] S. Arora, C. Lund, R. Motawani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998. 5

[3] S. Arora and S. Safra. Probabilistic checking of proofs : A new characterization of NP. *Journal of the ACM*, 45(1):70–122, 1998. 5

[4] A. Bogdanov and E. Viola. Pseudorandom bits for polynomials. *SIAM J. Comput.*, 39(6):2464–2486, 2010. 10

[5] J. Cai and A. Nerurkar. Approximating the SVP to within a factor $(1 + 1/\dim^{\epsilon})$ is NP-hard under randomized reductions. *Journal of Computer and Systems Sciences*, 59(2):221–239, 1999. 3

[6] Q. Cheng and D. Wan. Complexity of decoding positive-rate reed-solomon codes. In *ICALP*, pages 283–293, 2008. 1, 2, 3

[7] Q. Cheng and D. Wan. A deterministic reduction for the gap minimum distance problem. In *Proceedings of the ACM Symposium on the Theory of Computing*, pages 33–38, 2009. 1, 2, 3

[8] I. Dumer, D. Micciancio, and M. Sudan. Hardness of approximating the minimum distance of a linear code. In *Proc.* 40^{th} *IEEE Symposium on Foundations of Computer Science*, 1999. 1, 2, 3

[9] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43(2):268–292, 1996. 5

[10] I. Haviv and O. Regev. Tensor-based hardness of the shortest vector problem to within almost polynomial factors. In *Proceedings of the ACM Symposium on the Theory of Computing*, pages 469–477, 2007. 3

[11] S. Khot. Hardness of approximating the shortest vector problem in high L_p norms. In *Proc. 44th IEEE Symposium on Foundations of Computer Science*, 2003. 3

[12] S. Khot. Hardness of approximating the shortest vector problem in lattices. In *Proc. 45th IEEE Symposium on Foundations of Computer Science*, pages 126–135, 2004. 3

[13] S. Lovett. Unconditional pseudorandom generators for low degree polynomials. *Theory of Computing*, 5(1):69–82, 2009. 10

[14] D. Micciancio. The shortest vector problem is NP-hard to approximate to within some constant. *SIAM Journal on Computing*, 30(6):2008–2035, 2000. 3

[15] A. Vardy. The intractability of computing the minimum distance of a code. *IEEE Transactions on Information Theory*, 43(6):1757–1766, 1997. 2

[16] E. Viola. The sum of d small-bias generators fools polynomials of degree d. *Computational Complexity*, 18(2):209–217, 2009. 2, 10